On Self-Routing in Clos Connection Networks

Barry G. Douglass, *Member, IEEE,* and A. Yavuz Oruç, *Senior Member, IEEE*

***Abstract*—A self-routing connection network is a switching device where the routing of each switch can be determined in terms of the destination addresses of its inputs alone, i.e., independent of the routing information regarding the other switches in the network. One family of connection networks that were considered in the literature for self-routing are Clos networks. Earlier studies indicate that some Clos networks can be self-routed for certain permutations. This paper proves that the only category of Clos networks that can be self-routed for all permutations are those with at most two switches in their outer stages.**

I. Introduction

THIS paper considers the self-routability of Clos networks. Such networks find applications as connectors in telephone switching and interprocessor communications in parallel computers [15], [10], [5], [9] and have been the focus of much research. An n-input Clos network [4] where $n = mk$ for some positive integers m and k, consists of three stages as depicted in Fig. 1. The first and third stages of the network, each, consist of $n/m\, m \times m$ switches which may be implemented as crossbars, or by other smaller Clos networks and this may be recursively repeated. The center stage consists of $m\, n/m \times n/m$ switches which are similarly implemented. The network has exactly one link between every two switches in its consecutive stages. Throughout the paper the switches in the first stage will be denoted IS_i, those in the third stage will be denoted OS_i where $1 \leq i \leq n/m$, and the switches in the center stage will be denoted MS_i where $1 \leq i \leq m$. The network itself will be denoted $CL(n, m)$.

The networks we consider in this paper are all Clos networks, but possibly with different values of m. Two Clos networks will be of particular interest: the Clos network with $m = 2$ which is widely referred to as the Benes network in literature [1], and that with $m = n/2$, which is called the complementary Benes network [14], [3], [7].

Let Σ_n denote the set of all $n!$ permutation maps over a set of n elements. It is known that an n-input Clos network can realize each of these $n!$ permutations [1]. This means that, for each permutation $p \in \Sigma_n$, the network contains a set of disjoint paths between its inputs and outputs so as to connect input i to output $p(i); i = 1, 2, \cdots, n$. While this fact guarantees that a Clos network exhibits a set of disjoint paths corresponding

Paper approved by the Editor for Routing and Switching of the IEEE Communications Society. This work is supported in part by the National Science Foundation under Grant CCR-8708864. This paper was presented at the International Conference on Parallel Processing, St. Charles, IL, August 1990.

B. G. Douglass is with the Department of Electrical Engineering, Texas A&M University, College Station, TX 77843.

A. Y. Oruç is with the Department of Electrical Engineering and the Institute of Advanced Computer Studies, University of Maryland, College Park, MD 20742.

IEEE Log Number 9203585.

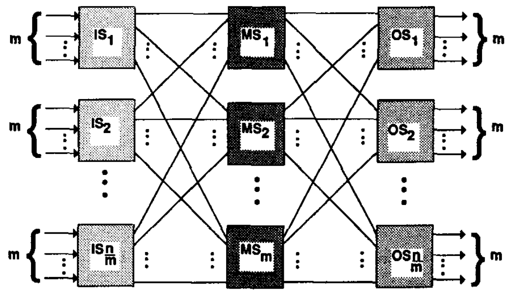


Fig. 1. An n-input Clos network.

to each permutation in Σ_n, it does not show how such a set of vertex disjoint paths can be formed. This problem is commonly referred to as the *network control,* or *network routing* problem, and has been posed in two different ways in the literature. The first approach, called *global routing,* assumes that there is a single controller (some kind of program or procedure) which receives the entire permutation p as input, and computes the settings for the switches in the network directly from this information. In contrast, the second approach, which is called *self-routing,* divides the information about the permutation p over the switches in some particular way, and this information can be transmitted only over the paths which exist between the switches. More precisely, a switching network is called *self-routing* if each of its switches can determine its setting only from the destination addresses of its own inputs, regardless of the destination addresses of the inputs of other switches in the network.

Much work has been reported on global routing schemes for Clos networks, both in sequential and parallel algorithm domains (see, for example, [13], [12], [8]). We shall not deal with such routing schemes in this paper. It suffices to say that global routing either requires too much time or too much hardware. For example, on a single processor, the n-input Benes network needs $O(n \log_2 n)$ steps to program and this is incompatible with the network's $O(\log_2 n)$ path length [?], [13]. The programming time can be reduced to $O(\log_2^2 n)$, by using an n-processor parallel computer, but this requires interconnection networks which cost more than the Benes network itself [11], [8], [3].

These problems with global routing have prompted some research on the self-routing aspect of Clos networks. Nassimi and Sahni established that many permutations, frequently used in parallel computations, and which they named *class F,* can be self-routing through the Benes network [12]. Recently, Boppana and Raghevandra showed that many more permutations, which they named *class L,* can also be self-routed by the

0090-6778/93$03.00 © 1993 IEEE

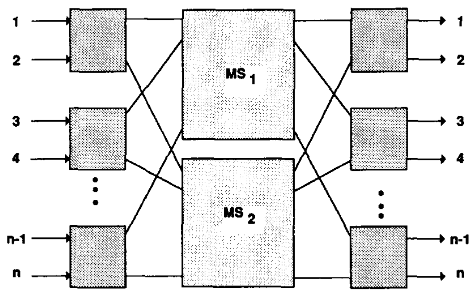


Fig. 2. The Benes network.

same network [2]. Most recently, it was shown in [7] that the complementary Benes network can be self-routed.

Given these results, a question naturally arises as to whether Clos networks can self-route all permutations. The main result of this paper is the answer to this question in the negative. In Section II, it is shown that all Benes networks with more than five inputs are not self-routing. More generally, in Section III, it is shown that no Clos network whose first stage contains more than two switches is self-routing. The paper is concluded in Section IV.

II. Self-Routability of Benes Networks

First, consider the self-routability of the Benes network which is depicted in Fig. 2. It is obvious that the Benes networks with one and two inputs are both self-routing. Some additional thought reveals that the Benes networks with three and four inputs are also self-routing. Furthermore, given that the 4-input Benes network is self-routing, it is easy to see that the 5-input Benes network is also self-routing if its fifth input and output are connected to both switches in the center stage.

We prove that these are the only Benes networks which can be self-routed. The proof is carried out by first observing that the number of inputs which are mapped to each half of outputs in such a network is constant over all permutations in Σ_n. We then classify the types of switches in the first stage according to how their inputs are mapped to each half of outputs under permutations in Σ_n. Next, we prove that for all even $n \geq 8$, there exist permutations in Σ_n which, when modified in a certain way, force the number of inputs mapped to the two halves of outputs through a center-stage switch to change. This then leads to the proof that the Benes network is not self-routing for all even $n \geq 8$. What remains to be considered is the case for $n = 6$, and networks with odd numbers of inputs both of which we will handle separately.

Let R_1 denote the set of outputs of the first $\lceil n/4 \rceil$ switches and R_2 denote the set of outputs of the next $\lfloor n/4 \rfloor$ switches in the third stage of an n-input Benes network. Let $\mu^p_{i,j}$ denote the number of inputs which are mapped into R_j by permutation p through center-stage switch MS_i where $1 \leq i, j \leq 2$.

Proposition 1: $\mu^p_{i,1} = \lceil n/4 \rceil$ and $\mu^p_{i,2} = \lfloor n/4 \rfloor$ for all $p \in \Sigma_n, n$ even, and $1 \leq i \leq 2$.

Proof Each center stage switch has $n/2$ output links, one to each of the $n/2$ third stage switches. Exactly $\lceil n/2 \rceil$ of these are connected to $OS_1, OS_2, \cdots, OS_{\lceil n/4 \rceil}$ whose outputs define R_1, and the rest $\lfloor n/4 \rfloor$ to $OS_{\lceil n/4 \rceil+1}, OS_{\lceil n/4 \rceil+2}, \cdots, OS_{n/2}$ whose outputs define R_2. Since all n output links between the center and third-stage switches must be occupied to realize any permutation in Σ_n, each $p \in \Sigma_n$ must map $\lceil n/4 \rceil$ inputs into R_1 and $\lfloor n/4 \rfloor$ inputs into R_2 through the $n/2$ output links of each of the center-stage switches. □

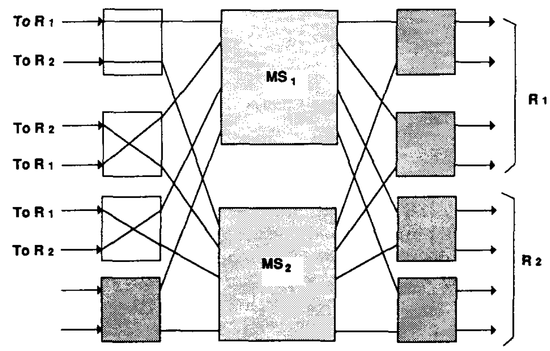


Fig. 3. Type 3 switches in a Benes network.

Let $S^p_{i,j}$ denote the set of switches in the first stage of an n-input Benes network one input of which is mapped to R_i and the other input of which is mapped to R_j by $p \in \Sigma_n$ where $1 \leq i \leq j \leq 2$. Call the switches for which $i = j = 1$, switches of *type 1,* those for which $i = j = 2$, switches of *type 2,* and those for which $i = 1, j = 2$, switches of *type 3.* Then the following statements hold.

Proposition 2: For all even $n \geq 8$ there exists a permutation $p \in \Sigma_n$ for which $|S^p_{1,2}| \geq 3$.

Proof: For $n \geq 8$, the first stage contains at least four switches. Choose p so that at least three of these switches belong to $S^p_{1,2}$. Obviously, p is not unique. □

Proposition 3: For all even $n \geq 8$ there exists a permutation in Σ_n which maps one input of each of at least two first-stage switches of type 3 to R_1 through the same center-stage switch, and the other inputs of these two switches to R_2 through the other center-stage switch.

Proof: From Proposition 2, there exists at least four first stage-switches any subset of which can be fixed as type 3 switches by choosing an appropriate permutation in Σ_n. If at least three are fixed as type 3 switches, then, obviously, that permutation must map one input of each of at least two of these switches into R_1 through the same center-stage switch, and their other inputs into R_2 through the other center-stage switch. ||

A graphical construction of this proposition is depicted in Fig. 3. It is seen that the first three switches in the first stage are fixed as type 3, and the first two map one of their inputs to R_1 through the upper center-stage switch, and their other inputs to R_2 through the lower center-stage switch. The third switch routes one of its inputs to R_1 through the lower center-stage switch and its other input to R_2 though the upper center-stage switch. The fourth switch is left unspecified, even though it should also be of type 3 as implied by the following proposition.

Proposition 4: $|R_i| = 2|S^p_{i,i}| + |S^p_{1,2}|; 1 \leq i \leq 2$, and $|S^p_{1,1}| - |S^p_{2,2}| = \lceil n/4 \rceil - \lfloor n/4 \rfloor$ for all $p \in \Sigma_n$.

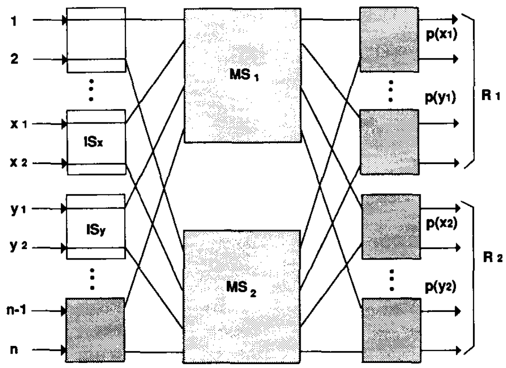


Fig. 4. The partial setting of a Benes network under permutation p.

Proof: That $|R_i| = 2|S^p_{i,i}| + |S^p_{1,2}|; 1 \le i \le 2$ is obvious since each first-stage switch of type 1 as well as type 2 contribute two inputs, and each first-stage switch of type 3 contributes one input to R_i. That $|S^p_{1,1}| - |S^p_{2,2}| = \lceil n/4 \rceil - \lfloor n/4 \rfloor$ follows immediately from the fact that $R_i = 2|S^p_{i,i}| + |S^p_{1,2}|; 1 \le i \le 2$, and $R_1 = 2\lceil n/4 \rceil$, and $R_2 = 2\lfloor n/4 \rfloor$. □

We now state the main result of this section.

Theorem 1: No Benes network with an even number of inputs equal to, or greater than eight can be self-routed.

Proof: First, by Proposition 2, construct a permutation p for which $|S^p_{1,2}| \ge 3$. Then by Proposition 3, we can find two first-stage switches of type 3, say IS_x and IS_y such that p maps one input from each to R_1 through the upper center-stage switch, and the other input from each to R_2 through the lower center-stage switch as shown in Fig. 4. Call the inputs to the first of these two switches x_1, x_2, and the inputs to the second y_1, y_2, so that $p(x_1), p(y_1) \in R_1$, and $p(x_2), p(y_2) \in R_2$. Now construct another permutation, say $q \in \Sigma_n$ such that $q(x_2) = p(y_1)$ and $q(y_1) = p(x_2)$, and $q(x) = p(x)$ for all other inputs x over which p is defined, i.e., the remaining inputs of the network. Since only the outputs of x_2 and y_1 have changed under q, the network can accomodate this change only by redefining the states of the two switches to which these inputs are connected if it is to be self-routing. However, regardless of how the two switches are set, now both inputs to both switches are to be mapped to the same set of outputs, i.e., both inputs of IS_x to R_1, and both inputs of IS_y to R_2. Therefore, the number of inputs which are routed to R_1 through the upper center-stage switch decreases by one, and the number of inputs which are routed to R_1 through the lower center-stage increases by one. A similar change occurs in the number of inputs which are routed to R_2. But this contradicts Proposition 1 which states that the number of inputs which are mapped to each half of outputs of a Benes network through each of its center-stage switches is constant for all $p \in \Sigma_n$. Hence the statement follows. □

We now consider the remaining cases. For $n = 6$, the Benes network has three switches in each of its outer stages and two switches in the center stage. In order for the network to be self-routing, the setting of each switch must be fixed for each pattern of inputs it receives. In Fig. 5 we list a sequence of permutations and their realizations on a 6-input Benes network. In Fig. 5(a), switch IS_1 is arbitrarily fixed so that input 1 is routed to output 4 through switch MS_1, and input 2 is routed to output 1 through MS_2. The settings of switches IS_2 and IS_3 are then determined. (In fact, IS_3 can be set either way, but this does not alter our argument.) In Fig. 5(b), the destinations of IS_1 and IS_3 have changed, and the destination of the inputs of IS_2 are kept the same as in Fig. 5(a) as highlighted by shading. Therefore, the setting of IS_2 remains the same and the settings of the other switches are then determined. Continuing in the same way, we arrive at Fig. 5(d), and find that the setting of IS_1 disagrees with that in Fig. 5(a) even though the destinations of its inputs remain the same. We conclude that the network is not self-routing.

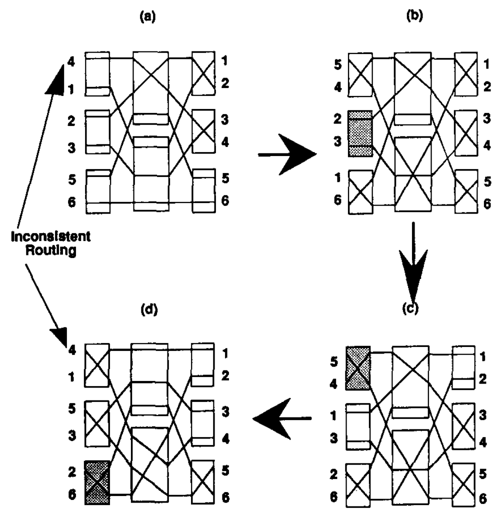


Fig. 5. A sequence of routings for the 6-input Benes network.

As for odd values of $n \ge 7$, it is easily seen that a Benes network with an odd number of inputs can be obtained by adding an input and output to the Benes network with $n - 1$ inputs, and connecting them to both of the center-stage switches. The proof that a network with odd number of inputs is not self-routing is then an immediate consequence of the proof that the network with one less inputs is not self-routing.

Thus, we have established the following.

Theorem 2: An n-input Benes network is self-routing if and only if $n \le 5$.

III. Self-Routability of Clos Networks

Theorem 1 can be extended to Clos networks as follows. First, we divide the outputs into two halves R_1 and R_2 as before, and extend the definition of a type 3 switch as one which sends exactly $m/2$ inputs to R_1, and $m/2$ inputs to R_2. Then we note that Proposition 2 still holds for the extended type 3 switch. Therefore, for $n/m \ge 4$, we can construct a permutation, say p, so as to have at least three type 3 switches. Furthermore, along the lines of Proposition 3, we can find a switch, say MS_i, in the center stage such that p sends one input of each of at least two type 3 switches to R_1 through

MS_i. With these statements, we can now prove the following statement.

Theorem 3: The $CL(n,m)$ network with $n/m \geq 4$, and even $m \geq 2$ is not self-routing.

Proof: As specified above, choose a permutation, p so that MS_i receives one input from a first-stage switch IS_x of type 3 and one input from another first-stage switch IS_y of type 3 both going to R_1. Now, by interchanging the inputs of these two switches only, construct a permutation q under which all inputs of IS_x go to R_1, and all inputs of IS_y go to R_2, and which, otherwise, is identical to p. With this change, regardless of how the inputs of the two switches are routed, MS_i must send one input from IS_x to R_1, and one input from IS_y to R_2. But, since the destinations of the inputs of all switches except those of IS_x and IS_y remain unchanged under q, this requires that the number of inputs which are mapped to R_1 through MS_i decrease by one, and the number of inputs which are mapped to R_2 through MS_i increase by one. However, this contradicts the fact that the number of paths from MS_i to R_1 and R_2 is fixed, and hence the statement. $\square$

There are two cases which remain to be considered. That Clos networks with odd values of m cannot be self-routed immediately follows from the above theorem. The case for $n/m = 3$ can be proven by constructing a counter example which is analogous to that in Fig. 5, and is omitted here.

Combining the above statements we now have the following theorem.

Theorem 4: The $CL(n,m)$ network is self-routing if and only if $n/m \leq 2$, or $m = 1$.

IV. Conclusion

This paper has established that an n-input Benes network is self-routing if and only if $n \leq 5$. More generally, it has been shown that an n-input Clos network, with m-input switches in its outer stages, is self-routing if and only if $n/m \leq 2$, or $m = 1$. On a more positive note, the authors showed in another paper [6] that the same network can self-route at least $(m!)^{n/m}((n/m)!)^m$ permutations. For $m = \sqrt{n}$, this gives a network with $O(n^{1.5})$ cost and $O(1)$ delay that can self-route at least $(2\pi n^{1/2})^{n(1/2)}(n/e^2)^n$ permutations. It remains to be shown if this lower bound is tight, i.e., whether or not more permutations can be self-routed by the same network.

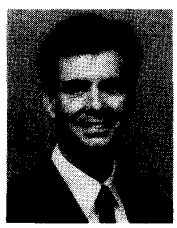
Along a different direction, in that same paper [6] the authors introduced a weaker form of self-routing which relies on balancing routes in Clos networks. In particular, it was shown that an n-input Benes network can be modified so as to have a network that can realize all permutations with $O(n \log_2^2 n)$ gates and in $O(\log_2^3 n)$ constant fan-in gate delays.

Acknowledgment

The authors thank the anonymous referees for their constructive remarks and suggestions.

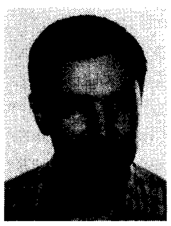
References

[1] V. E. Benes, *Mathematical Theory of Connecting Networks and Telephone Traffic.* New York: Academic, 1965.
[2] R. Boppana and C. S. Raghevandra, "On self-routing in Benes and shuffle-exchange networks," in *Proc. Int. Conf. Parallel Processing,* St. Charles, IL., Aug. 1988, vol. 1, pp. 196–200.
[3] J. D. Carpinelli and A. Y. Oruç "Parallel set-up algorithms for Clos networks," in *Proc. 2nd Int. Conf. Supercomputing,* Santa Clara, CA, May 1987, pp. 321–327.
[4] C. Clos, "A study of non-blocking switching networks," *Bell Syst. Tech. J.,* March 1953, pp. 406–424.
[5] T.-Y. Feng, "A survey of interconnection networks," *IEEE Comput.,* pp. 12–27, Dec. 1981.
[6] B. G. Douglass and A. Y. Oruç "Self-routing and route balancing in connection networks," in *Proc. Int. Conf. Parallel Processing,* St. Charles, IL, May 1990, pp. 331–337.
[7] D. M. Koppelman and A. Y. Oruç, "A self-routing permutation network," in *Proc. Int. Conf. Parallel Processing,* St. Charles, IL, 1989, vol. 1, pp. 288–295.
[8] G. F. Lev, N. Pippenger, and L. Valiant, "A fast parallel algorithm for routing in permutation networks," *IEEE Trans. Comput.,* vol. C-30, pp. 93–100, Feb. 1981.
[9] M. J. Marcus, "The theory of connecting networks and their complexity: A review," *Proc. IEEE,* vol. 65, pp. 1263–1271, Sept. 1977.
[10] G. M. Masson, G. C. Gingher, and S. Nakamura, "A sampler of circuit switching networks," *IEEE Comput.,* pp. 32–48, June 1979.
[11] D. Nassimi and S. Sahni, "Parallel algorithms to set up the Benes permutation network," *IEEE Trans. Comput.,* vol. C-31, pp. 148–154, Feb. 1982.
[12] ——, "A self-routing Benes network, and parallel permutation algorithms," *IEEE Trans. Comput.,* vol. C-30, pp. 157–161, May 1981.
[13] D. C. Opferman and N. T. Tsao-Wu, "On a class of rearrangeable switching networks," *Bell Syst. Tech. J.,* pp. 1579–1600, May–June 1971.
[14] D. S. Parker, Jr., "New points of view on three-stage rearrangeable switching networks," in *Proc. Workshop Interconnection Networks,* Computer Society Pub., 1980, pp. 56–63.
[15] N. Pippenger, "Telephone switching networks," in *Proc. Symp. Appl. Mathemat.,* vol. 26, May 1982, pp. 101–113.



Barry G. Douglass (S'89–M'90) received the B.S. degree in aeronautical engineering from Rensselaer Polytechnic Institute in 1972, the M.B.A. degree from the University of Texas at Austin in 1975, the M. S. degree in computer and systems engineering from Rensselaer Polytechnic Institute in 1987, and the Ph.D. degree in computer and systems engineering from Rensselaer Polytechnic Institute in 1989.

Since 1990 he has been an Assistant Professor in the Department of Electrical Engineering at Texas A&M University. His research interests include parallel computer architecture, interconnection networks, and intelligent systems.



A. Yavuz Oruç (S'81–M'81–M'83–SM'92) received the B.Sc. degree in electrical engineering from the Middle East Technical University, Ankara, Turkey, in 1976, the M.Sc. degree in electronics from the University of Wales, Cardiff, United Kingdom, in 1978, and the Ph.D. degree from Syracuse University, Syracuse, NY, in 1983.

Since January 1988, he has been an Associate Professor in the Department of Electrical Engineering at the University of Maryland, College Park, MD. Prior to joining the University of Maryland, he was on the faculty of the Department of Electrical, Computer and Systems Engineering at Rensselaer Polytechnic Institute, Troy, NY. His research interests include parallel computer and communication systems.

Dr. Oruç is a member of IEEE Computer and Information Theory Societies.